# A REAL-TIME DEFECT DETECTION IN PRINTED CIRCUIT BOARDS APPLYING DEEP LEARNING

Van-Truong Nguyen ⊠ Department of Mechatronics Engineering<sup>1</sup> nguyenvantruong@haui.edu.vn

*Huy-Anh Bui* Department of Mechatronics Engineering<sup>1</sup>

<sup>1</sup>Hanoi University of Industry 298 Cau Dien str., Bac Tu Liem District, Hanoi, Vietnam, 100000

Corresponding author

### Abstract

Inspection of defects in the printed circuit boards (PCBs) has both safety and economic significance in the 4.0 industrial manufacturing. Nevertheless, it is still a challenging problem to be studied in-depth due to the complexity of the PCB layouts and the shrinking down tendency of the electronic component size. In this paper, a real-time automated supervision algorithm is proposed to test the PCBs quality among different scenarios. The density of the PCBs layout and the complexity on the surface are analyzed based on deep learning and image feature extraction algorithms. To be more detailed, the ORB feature and the Brute-force matching method are utilized to match perfectly the input images with the PCB templates. After transferring images by aiding the RANSAC algorithm, a hybrid method using modern computer vision algorithms is developed to segment defective areas on the PCBs surface. Then, by applying the enhanced Residual Network-50, the proposed algorithm can classify the groove defects on the surface mount technology electronic components which minimum size up to 1×3 mm. After the training process, the proposed system is capable to categorize various types of overproduced, recycled, and cloned PCBs. The speed of the quality testing operation maintains at a high level with an average precision rate up to 96.29 % in case of good brightness conditions. Finally, the computational experiments demonstrate that the proposed system based on deep learning can obtain superior results and it outperforms several existing works in terms of speed, precision, and robustness.

Keywords: printed circuit board, defect detection, deep learning, computer vision, RESNET.

#### DOI: 10.21303/2461-4262.2022.002127

## 1. Introduction

Printed circuit boards (PCBs) [1–5] are a mandatory part of most modern electronic devices. Furthermore, the upward trend of consumer electronics products and the increasing demands for industrial electronic equipment have promoted applications of the accuracy of PCBs assembly operation. The circuit layout of a PCB presents a unique and single pattern in the image. Since the electronic device parts are shrinking down to minimize sizes, PCB is becoming more and more sophisticated and dedicated. Hence, detecting minor defects on the PCBs surface plays a significant role in the new era of intelligent industry. During manufacturing processes, numerous PCB defects can be engendered such as mistaken open circuits, missing components, causing the yield to drop, and component shifting.

The past decade has witnessed a huge growth in this PCB inspection field. One of the most widely used methods is the reference inspection method. In [6], the reference-free path-walking method is designed to detect component errors on high-density packages. However, it is only suitable for those with a ball grid array. The PCB reverse engineering through X-ray computed tomography (CT) is also proposed in [7]. Nevertheless, the X-ray CT framework needs a complicated configuration process to estimate exactly the separate sides of the PCB structure. Besides, performing the alignment of the reference images and the testing data precisely is the remarkable difficulty of this solution. In general, the image alignment could be solved by two well-known approaches: region-based match [8, 9] and feature-based match [10, 11]. The region-based approach directly uses the pixel values for the match. The similarity parameters are measured in detail by aiding the simple image subtraction, such as Sum of Squared Differences (SSD) [12], Sum of Absolute Differences (SAD) [13], and Normalized Cross Correlation (NCC) [14, 15]. Unfortunately, these methods are sensitive to transition and rotation changes. In addition, they require the repeat evaluation to test all possible geometric transformations of the input images, the computation load could be over-intensive. Meanwhile, the feature-based approach applied the edge points to calibrate the images continuously. It is called a straightforward feature representation. The Scale Invariant Feature Transform (SIFT) [16] and the RANdom SAmple Consensus (RANSAC) [17] are commonly used to search for the best edge points. Since these algorithms need a series of complicated processes to fit the matching rates, they are computationally very time constraints. To overcome these above drawbacks, Tsai and Chieh [18] offer an edge-based Expectation-Maximization algorithm to optimize the electronic part positions on the PCB surface. Even though the positioning precision could reach to a higher level, but the detected objects fail to indicate a minor geometric deviation with respect to the template inquiry.

With the development of deep-learning techniques, convolutional neural networks (CNN) [19-23] are also applied in the 4.0 manufacturing systems. Compared with traditional detection solutions, it has a plethora of merits such as high-speed detection, cost reduction, and good precision. In the literature review, the CNN-based models are incorporated with Support Vector Regression in [24]. Their study provides a powerful methodology to obtain the geometric transformation measurements of a test image without applying any manual search. Moreover, the subpixel accuracy of the images is boosted up rapidly within a few milliseconds, so this deep-learning model is appropriate for real-time implementation. The major weakness in their study is the intensity of the illumination. If the illumination condition is not well-controlled and changed terribly, the quality of the inspection procedure is seriously affected. In [25], an Improved Faster-RCNN and Feature Pyramid Network in run-time based on the Surface-mount technology (SMT) generation line, which belongs to the non-reference method is applied to deal with the illumination issues. In [26], the CNN based end-to-end network is developed in order to expand the size of the inputs. Additionally, they design two modules (the module of threshold estimation and the module of multi-label scoring) to enable multi-label classification in dysfunctional screen-printer. In [27], the architect ALexNet based on the CNN is discovered to locate six types of defects on the PCB surface (correct soldering, incorrect soldering, missing soldering, excess soldering, short circuit, and undefined object). The modified and compact tiny-YOLO-v2 network [28] is also integrated to classify various objects in a single picture of PCBs. Nonetheless, these approaches failed to control all characteristics of the same type of defects due to different materials and processes in PCB production. Hence, the detection performance was declined significantly in some negative cases. In addition, these techniques are highly complex and require several billion floating-point operations per second (FLOP) for a single pass over to achieve an image resolution of 224×224 in case of mimic components. As a result, the robustness against the noise of the system is declined rapidly.

In light of the remarkable importance and advantages previously mentioned, a real-time defect detection system is explored to ensure the quality of the PCBs surface. The proposed method is designed by coupling the ORB feature, the Brute-force matching method [29, 30] with the RANSAC algorithm. In particular, the PCB features are extracted and collated with the template data based on the ORB feature. The matching rates are adjusted by utilizing the Brute-force matching approach. Next, the geometric transformation of the input images is operated by aiding the RANSAC algorithm to locate defective areas on the PCBs surface. Then, the enhanced Residual Network-50 [31, 32] is developed and integrated with the control system to classify the groove defects. Compared to the existing works, the proposed approach has several advantages as follows:

1. By sorting in order a large number of layers and the Focal Loss function inside the Resnet-50 network, the proposed method is adaptable to point out the errors of the electronic components and check the component orientation. Besides, the proposed approach is consistent with classifying the mimic components which have the size of 1×3 mm.

2. The accumulation of the PCB defect database requires a short time for obtaining and updating. Hence, the self-calibrated ability of the proposed method is improved.

3. The proposed approach is applied on a real-time inspection system to ensure the PCBs quality. Besides, the display of the PCBs detailed information is built automatically based on the Graphical User Interface using PyQt. Thus, the workforce and the manual resources are limited steadily.

## 2. Materials and methods

Before operating the system, the users select the types of PCBs that need to be inspected. This pre-processing is complete automatically. After that, the core system operation is indicated as in **Fig. 1**. Overall, there are two main stages included in this process. The first stage is finding and creating ROI errors in PCBs. To be more detailed, this stage is performed over five sub-steps. To begin with, ORB features of the input images are extracted carefully. Then, these features are matched with the similar ORB features in the template data by aiding Bruce – Force matching method. As a result, the best edge key points will be found. By integrating those best key points with the RANSAC algorithm, the transform matrix is figured out. This matrix is used to calibrate the input images and the template data are converted to YUV color space and performed image subtraction. Next, the resulting images are filtered noise by using the opening method and threshold in order to create ROI errors of the PCBs. At the second stage, all types of ROI errors are classified based on the RESNET-50 models to output the names of the missing components or check the orientation of the components.



Fig. 1. Flowchart of the proposed algorithm

After receiving from the camera, the image data is calibrated to eliminate distortion which could affect the accuracy of the proposed algorithm. The difficult problem is matching perfectly the input one with the PCBs template images. It is dealt with this issue by establishing the most suitable transform matrix between two images. Assume that  $[u_i, v_i]$  is the coordinate of a point in the input image and  $[u'_i, u'_j]$  is the coordinate of a point in PCB template image, H is the homography transformation matrix. The 2D – form of the matrix H is defined as:

$$\begin{bmatrix} u'_{i} \\ v'_{i} \\ 1 \end{bmatrix} = H \cdot \begin{bmatrix} u_{i} \\ v_{i} \\ 1 \end{bmatrix}, \ H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}.$$
 (1)

The equation (1) is solved by applying the singular-value Decomposition algorithm [33]. It is clear that at least four pairs of the points are utilized to calculate the matrix H. To find four pairs of points which describe the image features, the ORB feature and Brute-force matching method [29, 30] is used. In particular, the ORB feature is adopted to search extracted key points from the two images. Brute-Force matcher is used to calculate the distance value between each pair of key points in two images.

After the calculation process, the matrix H is created. However, one major shortcoming of the distance value is treating the feature vectors equally. Hence, some pairs of points could be matched wrong and become outliers. To handle this drawback, the RANSAC algorithm [17] is applied. Basically, the RANSAC algorithm chooses random pairs of feature points to calculate the matrix H. The RANSAC starts with a simple model based on a small subsample of the data, then uses the remaining data points to identify consensus and outlier points. The outliers are removed, and the model is recomputed. Finally, the most suitable matrix H is established based on the result pairs of points. The input image is transformed into the template image as illustrated in Fig. 2.

The optimized samples of the RANSAC algorithm K are chosen by the formula below:

$$1 - \left(1 - \left(1 - \partial\right)^n\right)^K = \rho, \tag{2}$$

where  $\partial$  is the probability that a point is an outlier; *n* is the number of points in a single sample; *K* is the number of optimized samples;  $\rho$  is desired probability that a good sample is created.

After transforming the input images successfully, both the result images and the PCB template are converted into YUV color space to decline the impact of the luminance. Since then, they are called the YUV dataset. At the next step, the absolute difference of each channel in this YUV dataset is computed. Let's denote that I is a pixel value of the input images, T is a pixel value of the template images, c is one of three Y, U, V channels, E is absolute difference pixel values. The absolute difference is expressed as:

$$E = |I_c - T_c|. \tag{3}$$

The result images are transformed into binary images to segment error areas. The error areas are called a blob. Since the noise phenomena that appear in the image could decline the resolution and the quality of the results, the OPENING method [34] is applied. This method is designed based on two consecutive methods: Dilation and Erosion [35]. Therefore, the ratios of the distorted noise are eliminated by utilizing three kinds of thresholds with the blob detailed parameters as illustrated in the group of inequalities as follow:

$$\begin{cases} w \cdot h < Thresh_{area}, \\ \frac{w \cdot h}{s} < Threh_{solid}, \\ \left\{ \frac{w}{h}, \frac{h}{w} \right\} < Thresh_{scale}, \end{cases}$$
(4)

where w is the blob width, *h* is the blob height, *s* is the blob area.



a - PCB input image; b - PCB template image; c - Result image after transformation

The process of finding error areas is shown in Fig. 3.

The PCB defects can be generated in various production processes, such as missing values, lacking components, mistaken open circuits, and short circuits, causing the yield to drop. In this paper, our system concentrates on two popular types of errors on the PCBs surface: component identification and mimic components are reversed. Thus, two deep learning models are used to complete these tasks. Specifically, RESNET-50 is the main backbone of the proposed system. RESNET model has the advantage of building a CNN model with hundreds or thousands of layers without being affected by the Vanishing Gradient problem. Vanishing Gradient problem occurs encountered when training deep learning model with gradient-based learning and backpropagation. In such a case, weights of the neural network are updated proportionally to the partial derivative of the loss function with respect to the current weight in each loop of the training process. However, if the gradient value vanishing is small, it may prevent the weight from changing its value. In some cases, this problem can completely end the deep learning model from deeper training. In order to solve this problem, the RESNET model uses Residual block. It includes two 3×3 convolutional layers with the same number of output channels and skips connections, or shortcuts to jump over layers.



**Fig. 3.** The process of defect detection in PCB board: a – Input image; b – Transformed image; c – YUV dataset; d – Absolute difference image after converting into binary image; e – image after using OPENING; f – Result image with blob after applying three kinds of thresholds

In this paper, the enhanced RESNET-50 network is applied with 50 layers. As mentioned above, there are two RESNET-50 models integrated in the proposed system. The first model is used for classifying the name of missing components and the second one is used for checking the orientation of the components. The input of the first model is the error areas detected in the previous step which is resized into  $300 \times 300$  pixels. The output of the model is the name of the electronic components. In the second model, the input is pre-defined components which need to be detected the orientation such as a microchip, a capacitor, etc. The output of this model is the respective orientation of the components. If these values are similar to the predefined orientation of the components, the components are correct. In contrast, the orientation checking process is conducted continually until reaching the best results.

The loss function plays a vital role in the retention of precision rate in a deep learning network. In a normal classification network, Cross-entropy loss is usually utilized in semantic segmentation tasks. The equation of the Cross-entropy loss is expressed as:

$$E(p_t) = -\sum_{n=1}^{N} \sum_{k=1}^{K} \sum_{j=1}^{P} d_{nkwh} \log(p_t),$$
(5)

where as N, K and P are the mini-batch size, the number of classes and number of pixels, respectively;  $d_{nkwh}$  is a one-hot vector and  $d_k = 1$  when k is a true label;  $p_t$  is the model estimated probability for the class.

However, there has been a huge bias in the number of each electronic component image, so the normal loss function such as Binary cross-entropy or categorical cross-entropy could not be adaptable to maintain the high accuracy of the models. For instance, the dataset might have 2000 images of resistors but there are only 100 push button images. Thus, the Focal loss [36] is applied in our mentioned models. This solution improves on addressing class imbalance during the training process. To be more detailed, the Focal Loss calculates a modulating term to cross-entropy

loss, so it is called the scaled cross-entropy loss. By aiding the Focal loss, the models could easily down-weight the contribution of easy examples and focus more on learning among the negative cases. The equation of the Focal Loss is shown as follows:

$$FL(p_t) = -\sum_{n=1}^{N} \sum_{k=1}^{K} \sum_{j=1}^{P} d_{nkwh} (1 - p_t)^{\gamma} \log(p_t),$$
(6)

with  $\gamma$  is the negative focus parameter. If  $\gamma = 0$ , the Focal loss equals the Cross-entropy loss.

## 3. Results and discussion

**Fig. 4** presents the hardware configuration of the proposed system. To begin with, a 5 Mpx webcam is installed to take the PCBs pictures. The images are then inserted into the Jetson Nano Developer Kit. After that, the NVIDIA Jetson Nano Developer Kit B01 is applied as the main core to perform the image processing algorithms. This kit allows running multiple Neural Networks in parallel mode for applications such as image classification, object detection, segmentation, and speech processing. Moreover, it processes several high-resolution sensors simultaneously. All platforms are the friendly user that operates at as little as 5 Watt. The Jetson Nano offers 472 GFLOPS with a quad-core 64-bit ARM CPU, an integrated 128-core NVIDIA GPU, and 4 GB LPDDR4 memory. At the next step, the input images are converted into the analog signal and transmitted to the Arduino Mega. The Arduino Mega 2560R3 is the center controller for this system.

The electromechanical system includes five main parts: the control box, the convey belts, the images processing box, the picking manipulator, and the HMI display monitor. The conveying system is divided into two separate convey belts. The first conveyor belt is used to move PCBs into the images processing box. Whereas the mission of the second conveyor belt is conducted the PCBs classification. The images processing box includes a camera and embedded computer JETSON NANO. Besides, the lighting sub-system is set up in the images processing box to optimize the classification accuracy. The picking manipulator is operated by a stepper motor, a pneumatic cylindrical, and vacuum equipment. If a PCB is detected as a failure, the manipulator picks it out of the convey belt. Finally, the HMI observation screen integrated with the Graphical user interface (GUI) displays detailed information for users to control the quality of the PCBs. The layout of the electromechanical system is shown in **Fig. 5**.



Fig. 4. The hardware structure of the proposed system



Fig. 5. The outside of the proposed system.

The designed system contains two mentioned RESNET-50 models. The first model is utilized to classify the component errors. In the first model, the dataset is trained with 7800 images of more than 20 different types of electronic components such as resistor, ceramic capacitor, main chips, etc. Meanwhile, the second model is applied to inspect the orientation of the PCBs components. The second model dataset includes 15000 component images with different orientations. The input dataset is shown in **Fig. 6**. The configuration of the local computer is as follows: Windows 10 Pro 64-bit, Intel® Core<sup>TM</sup> i5-8500 CPU @ 3 GHz (6 CPUs), and 8 GB RAM. After finishing the training process, the highest precision rate of the first model is about 99.12 % on the validation dataset while the proportion for the second model could peak at 98.34 %.

To present the detailed information on the PCBs, let's develop a display application based on PyQT Open Library. After capturing by the camera system, the initial PCBs images are shown on the designed monitor. Then, the processing results are obtained and added to the data fields of the main window for the purpose of evaluating the PCBs quality. The main display interface is demonstrated in **Fig. 7**.

78_8161_duoi_ph ai	11111111 111,99066,dusi, phai	160,32501,tren,tr	200_90903_tren_tr ai	292,2H51,duoi, phai	101,72729,duoi, phai	178,87412,5mm,tr	100,90693,tren,tr	464,76707,durei, phai	0,570,chip,1	0_4140_chip_1	L1,thach,anh	1,2,chip,2	2017 1,3,5%	100	201 1,8,10	201 4 1,9,110	105 1,11,10
558_40100_duri_ phai	1111111 1111111 673,54717,tem,p Nel	677,66384,tren,p	746,30527,duoi, phai	770_33565_duoi_ phai	835,25734,duoi,t mi	111111111 809_9582_tren_ph ai	981,53953,tren,tr pi	965_60441_tren_tr ai	1.12, 10	1,15,thach,anh	1_16_tu_gom	1_15_tu_gom	1,21,tro	1,23,tu.gem	1,28,tran	1,29, tran	1,32,800
0 1005_30050_duni _phai	2222222 019_50355_tren_ physi	1041_10052_tren_ phai	LULLIUU M LULLIUU Tota 93205, duoi Jani	100, 200 here	1107_733_bottom	1117_363_bottom	1124_6_right	1192,47075,tren, trai	1.H.W	1,15,14	1,36,nguon	1,185,nguen	1_1235_chip_cam	2,0,chip,2	2,1,tro	2,4,110	331 25,tro
1202,42211, duei	1273, 56226, hong	1A ASZ L+ H17_93_right	1436,40369, tree,	1476,72256, duoi	1479,58918,6400	1439,41027,tren,	1490_43473_tren_	1510_27104_dvei	19 2.9_tro	2_10_tro	2,14,ehip,1	2_15_thech_enh	Z_16_tu_gom	2,19,thach_anh	103 2,21,tro	<b>E01</b> 2,22,50	2_23_tu_gom
				a									h				

Fig. 6. The data set of training models: a – the first training model; b – the second training model



Fig. 7. The display interface of the proposed system

The designed system is evaluated among the different scenarios: poor brightness condition, full brightness condition. In order to show the improvement in performance, the proposed approach is compared to the tiny YOLOv2 algorithm [28]. The evaluation criteria are divided into three main elements: the lack of the electronic components, the component orientation, and the defect detection of the PCBs surface.

The experiments are conducted with 3 types of PCBs: Adruino mega, Adruino Uno, and Adruino Uno SMD. To be more specific, the number of test boards is 1000 units per kind. Each board is tested 200 times in total. The results are demonstrated in the sub-sections below.

**Fig. 8–10** present the comparisons of the output results between the two mentioned approaches. In particular, when operating in good illumination, the possibility to inspect the passed PCBs is similar between the two methods as indicated in **Fig. 9**. Nevertheless, it is remarkable that the proposed method always maintains better performance compared with the tiny YOLOv2 in the case of detecting failed PCBs. For instance, in **Fig. 10**, the proposed approach could precisely detect all kinds of errors on the PCB surface such as two scratched 1x3 mm capacitors, one missing button, one missing main chip, and one reversed capacitor. In contrast, the tiny YOLOv2 not only shows the insufficiency of failed components but also displays wrong results.



Fig. 8. The display result of the failed PCB on the HMI monitor under poor brightness conditions: a - The proposed method; b - Tiny YOLOv2



Fig. 9. The display result of the passed PCB on the HMI monitor under poor brightness conditions: a - The proposed method; b - Tiny YOLOv2



Fig. 10. The display result of the failed PCB on the HMI monitor under full brightness conditions: a - The proposed method; b - Tiny YOLOv2

During the real-time PCB inspection process, let's analyze the impact of light on the proposed algorithm and collates the figures to the tiny YOLOv2 algorithm. The comparison results are illustrated in **Table 1**. Overall, the detection proportions of the proposed method among different types of PCBs are much higher compared to those of the tiny YOLOv2 algorithm. For instance, under a full lightness state, the proposed system presents superior stability with accuracy ups to 99.12 %. In contrast, the highest precision percentage of the tiny YOLOv2 algorithm is only 90.27 % for Adruino Uno SMD. Moreover, in case of poor brightness conditions, the precision rate of the proposed system still maintains good performance and shows clear merit over the tiny YOLOv2 algorithm. For example, the figures of the designed approach for Adruino Uno are 86.67 %, 85.65 %, and 84.33 %. Meanwhile, the proportions of the tiny YOLOv2 algorithm are much lower, at about 80.12 %, 78.08 %, and 77.53 %, respectively. Especially, the tiny YOLOv2 method could not identify the Adruino Mega surface because the size of some electronic components is smaller than 1×3 mm. As a result, it can't inspect the quality of mimic components. On the other hand, it is noticeable that the proposed system still detects the Adruino Mega errors with the precision rate peak at 83.48 % even when the light intensity is declined significantly. In conclusion, the most remarkable characteristic of the proposed system to emerge from the experimental results is that it is well-adjusted to be embedded in the 4.0 industrial manufacturing.

Because of the limited local computer configuration, the proposed system neglects some tiny cracks or scratches on the PCB surfaces. In future work, other AI inspection algorithms will be conducted to further improve this inspection.

### Table 1

The comparison results

		The p	proposed metl	ıod	The tiny YOLOv2 algorithm [28]				
Scenarios	Type of PCBs	The lack of the electronic components	Check- ing the component orientation	Defect detection of the PCBs surface	The lack of the electron- ic compo- nents	Checking the compo- nent orienta- tion	Defect detection of the PCBs surface		
Poor brightness condition	Adruino Mega	83.48 %	83.21 %	82.74 %	not identified	not identified	not identified		
	Adruino Uno	86.67 %	85.65 %	84.33 %	78.32 %	78.08 %	77.53 %		
	Adruino Uno SMD	90.06 %	89.78 %	88.65 %	80.12 %	78.76 %	78.29 %		
Full brightness condition	Adruino Mega	98.65 %	96.72 %	96.29 %	not identified	not identified	not identified		
	Adruino Uno	98.91 %	97.26 %	97.36 %	87.83 %	87.44 %	87.52 %		
	Adruino Uno SMD	99.12 %	98.34 %	98.22 %	90.27 %	90.08 %	89.23 %		

## 4. Conclusions

In this paper, a real-time defect detection approach is proposed to test the PCBs quality among different scenarios. The detailed structure of the designed method is calibrated based on deep learning image feature extraction algorithm. An innovative strategy, which is combined between ORB feature, Brute-force matching, and RANSAC algorithm, is devised. Moreover, the enhanced Resnet-50 network is embedded inside the system to identify the electronic components on the PCBs surface. Besides, the PCB information display is also developed by aiding the PyQT Library. In summary, our method has several advantages. To begin with, the technique is robust against noise. Secondly, an obvious advantage is applying the Resnet-50 network to speed up the computations. This holds true even after implementing the two mentioned models. Finally, the method can be applied to detect defects in automatic inspection in real-time.

### Acknowledgments

The authors would like to send sincere thanks to the Hanoi University of Industry for their financial support in the implementation of this study.

### References

 Zhang, Y. (2021). Improved Numerical-Analytical Thermal Modeling Method of the PCB With Considering Radiation Heat Transfer and Calculation of Components' Temperature. IEEE Access, 9, 92925–92940. doi: https://doi.org/10.1109/ access.2021.3093098

- [2] Khiabani, N., Chiang, C.-W., Liu, N.-C., Kuan, Y.-C., Wu, C.-T. M. (2021). An Ultrawide Ku- To W-Band Array Antenna Package Using Flip-Chipped Silicon Integrated Passive Device With Multilayer PCB Technology. IEEE Microwave and Wireless Components Letters, 31 (7), 861–864. doi: https://doi.org/10.1109/lmwc.2021.3074940
- [3] Rajabzadeh, M., Ungethum, J., Herkle, A., Schilpp, C., Becker, J., Fauler, M. et. al. (2021). A PCB-Based 24-Ch. MEA-EIS Allowing Fast Measurement of TEER. IEEE Sensors Journal, 21 (12), 13048–13059. doi: https://doi.org/10.1109/jsen.2021.3067823
- [4] Llamazares, A., Garcia-Gracia, M., Martin-Arroyo, S. (2021). Characterization of Parasitic Impedance in PCB Using a Flexible Test Probe Based on a Curve-Fitting Method. IEEE Access, 9, 40695–40705. doi: https://doi.org/10.1109/access.2021.3064190
- [5] Huang, C.-M., Wang, S.-H., Wu, T.-Y., Huang, M.-C., Wu, R.-B. (2021). Systematic Design for Mitigation of RF Desense by Interleaved Power Line in Two-Layer PCB. IEEE Transactions on Components, Packaging and Manufacturing Technology, 11 (5), 859–864. doi: https://doi.org/10.1109/tcpmt.2021.3063321
- [6] Jin, W., Lin, W., Yang, X., Gao, H. (2017). Reference-Free Path-Walking Method for Ball Grid Array Inspection in Surface Mounting Machines. IEEE Transactions on Industrial Electronics, 64 (8), 6310–6318. doi: https://doi.org/10.1109/ tie.2017.2682008
- [7] Botero, U. J., Ganji, F., Asadizanjani, N., Woodard, D. L., Forte, D. (2020). Semi-Supervised Automated Layer Identification of X-ray Tomography Imaged PCBs. 2020 IEEE Physical Assurance and Inspection of Electronics (PAINE). doi: https:// doi.org/10.1109/paine49178.2020.9337738
- [8] Xia, S., Wang, F., Xie, F., Huang, L., Wang, Q., Ling, X. (2021). An Efficient and Robust Target Detection Algorithm for Identifying Minor Defects of Printed Circuit Board Based on PHFE and FL-RFCN. Frontiers in Physics, 9. doi: https:// doi.org/10.3389/fphy.2021.661091
- [9] Zhang, B., Yang, H., Yin, Z. (2015). A Region-Based Normalized Cross Correlation Algorithm for the Vision-Based Positioning of Elongated IC Chips. IEEE Transactions on Semiconductor Manufacturing, 28 (3), 345–352. doi: https://doi.org/ 10.1109/tsm.2015.2430453
- [10] Romero Subirón, F., Rosado Castellano, P., Bruscas Bellido, G., Benavent Nácher, S. (2018). Feature-Based Framework for Inspection Process Planning. Materials, 11 (9), 1504. doi: https://doi.org/10.3390/ma11091504
- [11] Zhou, Z., Wang, M., Cao, Y., Su, Y. (2020). CNN Feature-Based Image Copy Detection with Contextual Hash Embedding. Mathematics, 8 (7), 1172. doi: https://doi.org/10.3390/math8071172
- [12] Ghaffari, A., Fatemizadeh, E. (2018). Image Registration Based on Low Rank Matrix: Rank-Regularized SSD. IEEE Transactions on Medical Imaging, 37 (1), 138–150. doi: https://doi.org/10.1109/tmi.2017.2744663
- [13] Chandran, K. R. S., Chandramani, P. V. (2019). Energy-efficient system-on-chip reconfigurable architecture design for sum of absolute difference computation in motion estimation process of H.265/HEVC video encoding. Concurrency and Computation: Practice and Experience, 34 (8). doi: https://doi.org/10.1002/cpe.5461
- [14] Zhang, G., Kuang, Z., Wei, S., Huang, K., Liang, F., Yang, C.-F. (2018). Hardware Implementation for an Improved Full-Pixel Search Algorithm Based on Normalized Cross Correlation Method. Electronics, 7 (12), 428. doi: https://doi.org/10.3390/electronics7120428
- [15] Zhang, H., Song, A. (2017). A map-based normalized cross correlation algorithm using dynamic template for vision-guided telerobot. Advances in Mechanical Engineering, 9 (9), 1–12. doi: https://doi.org/10.1177/1687814017728839
- [16] Kaur, P., Pannu, H. S. (2018). Comprehensive review of continuous and discrete orthogonal moments in biometrics. International Journal of Computer Mathematics: Computer Systems Theory, 3 (2), 64–91. doi: https://doi.org/10.1080/23799927. 2018.1457080
- [17] He, H., Hu, Z., Wang, B., Luo, D., Lee, W.-J., Li, J. (2020). A Contactless Zero-Value Insulators Detection Method Based on Infrared Images Matching. IEEE Access, 8, 133882–133889. doi: https://doi.org/10.1109/access.2020.3011170
- [18] Tsai, D., Hsieh, Y. (2017). Machine Vision-Based Positioning and Inspection Using Expectation-Maximization Technique. IEEE Transactions on Instrumentation and Measurement, 66 (11), 2858–2868. doi: https://doi.org/10.1109/tim.2017.2717284
- [19] Edalatifar, M., Tavakoli, M., Setoudeh, F. (2021). A Deep Learning Approach to Predict the Flow Field and Thermal Patterns of Nonencapsulated Phase Change Materials Suspensions in an Enclosure. Journal of Applied and Computational Mechanics. doi: https://doi.org/10.22055/jacm.2021.37805.3092
- [20] Alzghoul, A., Jarndal, A., Alsyouf, I., Bingamil, A., Ali, M., AlBaiti, S. (2021). On the Usefulness of Pre-processing Methods in Rotating Machines Faults Classification using Artificial Neural Network. Journal of Applied and Computational Mechanics, 7 (1), 254–261. doi: https://doi.org/10.22055/jacm.2020.35354.2639
- [21] Gao, Q., Wu, X. (2021). Real-Time Deep Image Retouching Based on Learnt Semantics Dependent Global Transforms. IEEE Transactions on Image Processing, 30, 7378–7390. doi: https://doi.org/10.1109/tip.2021.3104173
- [22] Wang, Y., Ji, S., Zhang, Y. (2021). A Learnable Joint Spatial and Spectral Transformation for High Resolution Remote Sensing Image Retrieval. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 8100–8112. doi: https://doi.org/10.1109/jstars.2021.3103216

- [23] Attaran, B., Ghanbarzadeh, A. (2014). Bearing Fault Detection Based on Maximum Likelihood Estimation and Optimized ANN Using the Bees Algorithm, Journal of Applied and Computational Mechanics, 1 (1), 35–43. doi: https://doi.org/10.22055/ jacm.2014.10547
- [24] Tsai, D.-M., Chou, Y.-H. (2020). Fast and Precise Positioning in PCBs Using Deep Neural Network Regression. IEEE Transactions on Instrumentation and Measurement, 69 (7), 4692–4701. doi: https://doi.org/10.1109/tim.2019.2957866
- [25] Hu, B., Wang, J. (2020). Detection of PCB Surface Defects With Improved Faster-RCNN and Feature Pyramid Network. IEEE Access, 8, 108335–108345. doi: https://doi.org/10.1109/access.2020.3001349
- [26] Park, J.-Y., Hwang, Y., Lee, D., Kim, J.-H. (2020). MarsNet: Multi-Label Classification Network for Images of Various Sizes. IEEE Access, 8, 21832–21846. doi: https://doi.org/10.1109/access.2020.2969217
- [27] Zhang, Q., Liu, H. (2021). Multi-scale defect detection of printed circuit board based on feature pyramid network. 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA). doi: https://doi.org/10.1109/ icaica52286.2021.9498174
- [28] Adibhatla, V. A., Chih, H.-C., Hsu, C.-C., Cheng, J., Abbod, M. F., Shieh, J.-S. (2020). Defect Detection in Printed Circuit Boards Using You-Only-Look-Once Convolutional Neural Networks. Electronics, 9 (9), 1547. doi: https://doi.org/10.3390/ electronics9091547
- [29] Nguyen, V.-T., Nguyen, A.-T., Nguyen, V.-T., Bui, H.-A., Nguyen, X.-T. (2021). Real-time Target Human Tracking using Camshift and LucasKanade Optical Flow Algorithm. Advances in Science, Technology and Engineering Systems Journal, 6 (2), 907–914. doi: https://doi.org/10.25046/aj0602103
- [30] Nguyen, V.-T., Nguyen, A.-T., Nguyen, V.-T., Bui, H.-A. (2021). A Real-Time Human Tracking System Using Convolutional Neural Network and Particle Filter. Lecture Notes in Networks and Systems, 411–417. doi: https://doi.org/10.1007/ 978-981-16-2094-2\_50
- [31] Helwan, A., Sallam Ma'aitah, M. K., Abiyev, R. H., Uzelaltinbulat, S., Sonyel, B. (2021). Deep Learning Based on Residual Networks for Automatic Sorting of Bananas. Journal of Food Quality, 2021, 1–11. doi: https://doi.org/10.1155/2021/5516368
- [32] Liu, C., Zhou, W., Chen, Y., Lei, J. (2020). Asymmetric Deeply Fused Network for Detecting Salient Objects in RGB-D Images. IEEE Signal Processing Letters, 27, 1620–1624. doi: https://doi.org/10.1109/lsp.2020.3023349
- [33] Akritas, A. G., Malaschonok, G. I. (2004). Applications of singular-value decomposition (SVD). Mathematics and Computers in Simulation, 67 (1-2), 15–31. doi: https://doi.org/10.1016/j.matcom.2004.05.005
- [34] Tran, Q.-V., Su, S.-F., Nguyen, V.-T. (2020). Pyramidal Lucas Kanade-Based Noncontact Breath Motion Detection. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 50 (7), 2659–2670. doi: https://doi.org/10.1109/tsmc.2018.2825458
- [35] Ma, S.-Y., Khalil, A., Hajjdiab, H., Eleuch, H. (2020). Quantum Dilation and Erosion. Applied Sciences, 10 (11), 4040. doi: https:// doi.org/10.3390/app10114040
- [36] Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollar, P. (2020). Focal Loss for Dense Object Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42 (2), 318–327. doi: https://doi.org/10.1109/tpami.2018.2858826

Received date 29.10.2021 Accepted date 18.03.2022 Published date 31.03.2022 © The Author(s) 2022 This is an open access article under the Creative Commons CC BY license

*How to cite:* Nguyen, V.-T., Bui, H.-A. (2022). A real-time defect detection in printed circuit boards applying deep learning. *EUREKA: Physics and Engineering, 2, 143–153. doi: https://doi.org/10.21303/2461-4262.2022.002127*